A NOVEL CONTRACT SIGNATURE BASED ON KEY EXCHANGE

by

Aakanksha Saha (111CS0412)

A thesis submitted in partial fulfillment of the requirements for the

Degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

under the guidance of

Prof. Sujata Mohanty

Department of Computer Science and Engineering

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

ROURKELA-769008, ORISSA

Declaration of Authorship

I, Aakanksha Saha, declare that this thesis titled, "A Novel Contract Signature Scheme based on Key Exchange" and the work documented in it is done by me. I confirm that:

- This work is completely done by me while in candidature for a B.Tech degree at this Institute.
- Where any fragment of this document has earlier been submitted for a degree or any other qualification at this Institute or any other institution, this has been clearly cited.
- Where I have consulted the published work of others, this is always explicitly attributed.
- Where I have cited from the work of others, the source is always mentioned.
- Where the thesis is based on the project done by myself or in collaboration with others, I have clearly stated what was done by others and what I have contributed myself.

Signed:

_____

Date:

_____

Abstract

A contract signature is a particular form of digital multi-signature that only involves two signers. Contract signing plays a critical role in any business transaction, particularly in situations where the involved parties do not trust each other. One of the most significant concerns in exchange signatures is the fraudulent and unfair exchange, which occurs when one party gets the signature of another party without giving his own signature. In the view of these security concerns, this thesis presents a secure and fair contract signature scheme based on key exchange protocol. The security and protection of the proposed scheme is based on solving hard computational assumptions such as discrete logarithm problem (DLP). The proposed protocol is abuse-free. The proposed scheme targets to have lesser computational overhead and high-security features than existing scheme[1]. The proposed scheme has wide application in real life scenarios, such as in electronic cash system.

Keyword: Contract signature; fair exchange; nonrepudiation; unforgeability; discrete logarithm problem.

Acknowledgements

I would like to express gratitude to my project guide, Prof. Sujata Mohanty for believing in my ability. Her thoughtful perceptiveness has enriched my research work. The flexibility of work she has offered me has deeply encouraged me producing the research. I am thankful to all the professors, batch mates and friends at National Institute of Technology, Rourkela for their cooperation and help. My full dedication to the work would not have been possible without their blessings and moral support.

Aakanksha Saha

111CS0412

Certificate

04 May, 2015

This is to certify that the work in the thesis entitled **A Novel Contract Signature based on Key Exchange** by **Aakanksha Saha**, bearing Roll No. 111CS0412,, is a record of an original research work carried out by her under my supervision and guidance in partial fulfillment of the requisites for the award of the degree of Bachelor of Technology in Computer Science and Engineering. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Prof. Sujata Mohanty
Dept. of Computer Science and Engineering
National Institute of Technology
Rourkela - 769008

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the increase in the amount of information in the present technical scenario, the need for security has also mushroomed. Apart from the size and amount, the worth of the information has also rose. Like any other asset of an organization, company or even individual information is the most valuable asset of all. The decline in the remoteness of the information and the ever evolving Web Applications adds to the need for increasing the confidentiality of the information and making it secure. In the present day, most of the communication takes place on the insecure channel that makes the message vulnerable to multiple attacks and threats. Creating a secure channel is very expensive and unscalable. While passing information over an insecure channel such as Internet to a person we must provide four different primary security requirements such as[10, 11],

- Confidentiality - Ensures protection from unauthorized persons.
- Integrity - Ensures the consistency of data.
- Authentication - Ensures the true identity of a person or an originator of a message.
- Non-repudiation - - Ensures no denial of communication.

These are the primary security goals that a secure system must satisfy for a successful communication. Earlier confidentiality was the only significant concern. In the last few years the other security parameters such as integrity verification, user authentication, digital signatures, etc. have been considered relevant to the confidentiality factor.

1.1 Introduction to Cryptography

Until recently cryptography referred almost exclusively to *encryption*, which is the process of converting ordinary information (called plain text) into unintelligible text (called cipher text).*Decryption* is the reverse, in other words, moving from the unintelligible cipher text back to plain text. A cipher (or cypher) consists of two algorithms; one that encrypts the message and the reverse decrypts the message. The key is the controlling instance in both of the algorithms. It is a secret (ideally known only to the communicants), usually a short string of characters, which is needed to decrypt the cipher text. While Symmetric Key Cryptography (shared key) was used earlier, they are kind of obsolete now due to various advancements in the technological field and the introduction of different algorithms for checking primality. Now-a-days Asymmetric Key Cryptography is preferred everywhere[10, 11]. So we can classify the cryptography into two parts:

- Symmetric Key Cryptography (shared key)
- Asymmetric Key Cryptography (different key)

1.1.1 Symmetric Key Cryptography

In symmetric key cryptography, the sender of the message encrypts the message by using the key "k". After receiving, the receiver, then decrypts the same message using the same key "k". The assumption is based on the fact that, both the sender and receiver use a shared common key and the transmission of the cipher text, and the key is done using an insecure channel. This system is vulnerable to threats if the key "k" is leaked and it is known to an adversary[10].

1.1.2 Asymmetric Key Cryptography

To overcome this particular problem of the symmetric key cryptography, the public key cryptosystem is used, where everything is same except few exception. First of all, there are two keys instead of one, one public key and one private key. To send a message securely, the sender encrypts his message with the receiver's public key. To decrypt the message,

the receiver uses his private key. The concept of Public Key Cryptography gave rise to the significant concept of Digital Signature. Electronically digital signature is analogous to the traditional handwritten signature. The main purpose of the digital signature is to enable a person to sign some electronic document digitally. One would wish for these digital signatures to have the equivalent properties as traditional signatures: they should be easy to generate, easy to verify and yet difficult to forge. By using the private key for signing, and the public key for verifying, this notion was successfully achieved. With time, many other variations of public-key cryptosystems and digital signatures were proposed[10].

1.2 Digital Signature

A digital signature checks and verifies whether an electronic document is authentic or not[13]. Digital signatures are mostly used to identify electronic entities for on-line transactions. A valid digital signature gives user a reason to consider that the information or message was created by a known authentic sender, such that the sender cannot deny that he has sent the message and that the message was unaltered in transmission. A digital signature uniquely recognizes the originator of digitally signed data and also guarantees the integrity of the signed data against corruption and misuse. Digital signatures are commonly used for software distribution, authenticate on-line entities, verify the origin of digital data, detect forgery attack.[12] A common digital signature procedure is shown in Figure 1.1 [13].

1.3 Contract Signature

Contract signing plays a critical role in any business transaction, particularly in situations where the involved parties do not trust each other. For instance, both the individuals sign two hard copies of the same contract at the same time and at the same place. After that, each keeps one copy as a valid and legal document that shows both of them have made commitments to the contract. If any one of them does not abide by the contract, the other one could provide the signed contract to a judge in the court. Cryptographic digital signatures have been used to provide non-repudiation services, such as signing a business

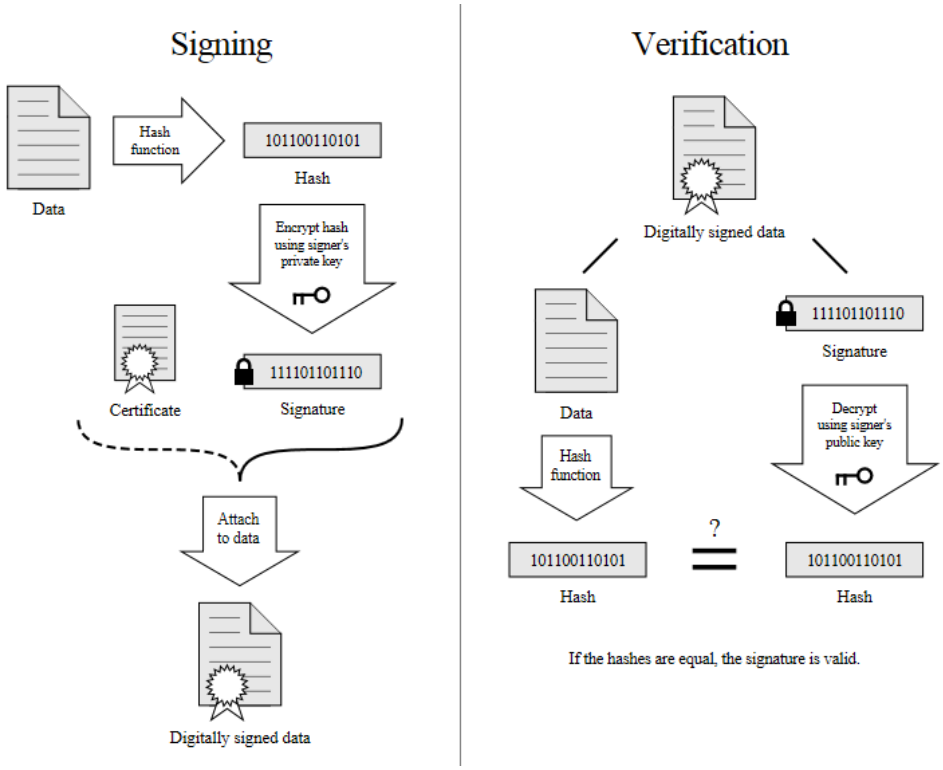

Figure 1.1: Digital Signature

contract. One class of multi-party protocols in the coming days can be the exchange of digital signatures. One of the most significant concerns in exchange signatures is the fraudulent and unfair exchange, which occurs when one party gets the signature of another party without giving his own signature[1, 4]. The notion of fairness is well established in a traditional message exchange, in which all parties involved obtain secrets simultaneously. However, in a network environment, secrets are exchanged over a network that does not provide a simultaneous exchange of messages, and, therefore, adequate cryptographic protocols are needed to facilitate secrets exchange. Earlier protocols were based on the gradual exchange of their secrets bit-by-bit in an interleaving manner. This process continues until both of them have received and released all the bits; but only after the end of the protocol, both the individuals can discover that the received bits are real secrets and are not erroneous[1, 4].

1.4 Our Contribution

In this thesis, we have contributed the following:

- We have proposed a new Contract Signature Scheme based upon key Exchange. We have analyzed the security of the scheme and also implemented it.
- We have compared and tabulated the performance efficiency of the existing scheme [1] and the proposed contract signature scheme.

1.5 Motivation

The motivation for this project came from the growing need for a contract signature scheme that can assure maximum possible security from the existing schemes. In cases where the signature protocol is executed unsuccessfully, each of the two parties involved cannot show the validity of the intermediate results produced by the other party to the outsiders. This scheme also satisfies some desirable security features, such as, fairness, timely termination, compatibility, and high performance.

1.6 Objective

The objective of my work is to generate a fair and secure contract signature scheme so to allow the two parties namely, party A and party B, to interact with each other efficiently.

1.7 Organization of thesis

In Chapter 2, in this chapter we have given the mathematical preliminaries which have been used throughout the thesis.

In Chapter 3, we have given the literature survey which includes the review of a existing contract signature scheme and the implementation of Harn's Scheme.

In Chapter 4, we have proposed our new contract signature scheme. Also the security analysis and the implementation result has been given in this chapter.

In Chapter 5, we provide conclusion of this thesis and future research directions.

Chapter 2

Mathematical Preliminaries

Following basic notations, definitions and models are used throughout this thesis.

2.1 Notation and Terminology

All groups discussed in this thesis are assumed to be abelian. Groups of prime order have useful properties and are widely used in cryptography. All groups of prime order are cyclic.

A group G is said to be cyclic if there is an element $g \in G$, such that for each $g' \in G$, there is an integer a with $g' = g^a$. Such an element is called a generator of G. For any prime integer p , the field of integers modulo p is denoted by Z_p. The cyclic multiplicative group of nonzero elements in Z_p is denoted as Z_p*.

2.2 Discrete Logarithm Problem (DLP)

Particularly in abstract algebra and its applications, discrete logarithms are group theoretic analogues of ordinary logarithms. An ordinary logarithm $log(a, b)$ gives a solution for the equation $a^x = b$ over the real or complex numbers. Likewise, if g and h are elements of a finite cyclic group G then a solution x of the equation is called a discrete logarithm to the base g of h in the group G. Briefly, if G is a finite group, the problem discrete logarithm is the following computational problem: given elements α and β in G , determine an integer x such that, $\alpha^x = \beta$, provided that such an integer exists[9].

2.3 Computational Diffie-Hellman Problem

The Diffie-Hellman problem can be described as follows. Let G be a cyclic group of order q. If g is a generator of some group, preferably the multiplicative group of a finite field, and x,y are randomly chosen integers then CDH assumption states that, given (g, g^a, g^b) for any randomly chosen generator g and $a, b \in \{0, ..., q-1\}$ it is computationally infeasible to compute the value of g^{ab}[13].

2.4 The Integer Factorization Problem

There are many fast algorithms for multiplying two given large prime numbers. On the other hand, it is considerably difficult to find the prime factors if the product of two large primes is given. The perceptible difficulty of factoring large integers forms the foundation of some modern cryptographic algorithms. If quick factorization of large primes is possible, these algorithms would not be secure anymore[16].

2.5 Safe Primes

Their relationship with the strong primes is what makes them safe prime. By definition a prime q is said to be a strong prime if $q+1$ and $q-1$ both have large prime factors. For a safe prime, $q = 2p+1$, the integer p is a large prime factor. The importance of safe primes is realized when they are used in discrete logarithm-based techniques like Diffie-Hellman key exchange. If $2p+1$ is a safe prime, the multiplicative subgroup of numbers modulo $2p+1$ has a subgroup of large prime order. Safe primes are used to minimize the modulus[15].

Chapter 3

Literature Survey

A number of protocols have been proposed till date to achieve fair message exchange. Historically, the protocol design has been evolved from the two-party approach, without the involvement of any trusted third party (TTP), to the TTP-based approach, where fairness is guaranteed with the involvement of a TTP. A major disadvantage associated with the two-party approach is that many numbers of exchange rounds are required to ensure an acceptable level of fairness; thus the overhead of communication becomes very high. In addition, gradual secret release protocols require all participating parties to have approximately equal level computational power in order to guarantee fairness in message exchange. Otherwise, the party with greater computational capabilities can launch a brute-force attack after receiving the first few bits, and work out the remaining bits of his counterpart's message/secret[4].

A contract signature is a digital signature generated by multiple signers, jointly with their private keys [1]. Multi signature allows any number of signers to work together to create a digital multi-signature corresponding to a message [7]. The length of digital multi-signature is equal to the length of each signature, and the public key of multi-signature is equal to the multiplication of all public keys of signers. A contract signature is a special form of digital multi signature that only involves two signers [1].

Chen et al. [3] proposed the concept of concurrent signature, which allows two parties,

without TTP, to produce and exchange two ambiguous signatures which are vague for any third party until one of the parties release an extra piece of information (called keystone) . A Concurrent signature is very efficient and requires neither a TTP nor a high degree of interaction between individuals. Later, the concept of perfect concurrent signatures was established by [5] and subsequently, asymmetric concurrent signature was also proposed by [6].

3.1 Review of Harn's Scheme [1]

Lein Harn and Lin [1] proposed a notion of contract signature used in e-commerce applications. They proposed a contract signature protocol based on the discrete logarithm problem and assumption. The contract signature protocol adopts a digital multi-signature scheme in public-key cryptography to ensure fair signature exchange over a network. This proposed solution allows many signers of a contract signature to transfer their partial signatures that are entirely ambiguous for any third party (i.e.,1 out ∞ of ambiguity) to generate a valid contract signature. In case any signer releases the partial signature to others, the signer does not bind to the contract.

3.2 Protocol for construction of a Contract Signature

3.2.1 System set-up

Let A and B agree to sign a contract m over Internet. We assume that A's private key is x_A, public key is $y_A = g^{x_A} modp$, and B's private key is x_B, public key is $y_B = g^{x_B} modp$.

3.2.2 Exchange Protocol

Step 1: A randomly selects a secret k_A, and computes a public value $r_A = g^{k_A} modp$. r_A is sent to B.

Step 2: In the same way, B randomly selects a secret k_B, and computes a public value $r_B = g^{k_B} modp$. r_B is sent to A.

Step 3: A computes $r = r_A.r_B mod p$. A solves s_A such that $s_A = x_A h(m,r) - k_A r mod p - 1$. s_A is sent to B.

Step 4: B verifies $y_A^{h(m,r)} = r_A^r.g^{s_A} mod p$. If it is, then B computes $r = r_A.r_B mod p$, and solves s_B such that $s_B = x_B h(m,r) - k_B r mod p - 1$. s_B is sent to A.

Step 5: A verifies $y_B^{h(m,r)} = r_B^r.g^{s_B} mod p$. Both A and B can now compute the contract signature (r,s), where $s = s_A + s_B mod p - 1$, of the contract m.The contract signature (r, s) of the contract m can be verified by a verifier by checking $(y_A.y_B)^{h(m,r)} = r^r.g^s mod p$.

Theorem 1. The combination of two partial signatures (r_A, s_A) and (r_B, s_B), in above protocol is a valid signature.

Proof. The two partial signatures (r_A, s_A) and (r_B, s_B), satisfy

$$y_A^{h(m,r)} = r_A^r.g^{s_A} mod p, and \tag{3.1}$$

$$y_B^{h(m,r)} = r_B^r.g^{s_B} mod p. \tag{3.2}$$

Multiplying above two equations, we obtain

$$(y_A.y_B)^{h(m,r)} = (r_A.r_B)^r.g^{s_A,s_B} mod p \tag{3.3}$$

$$(y_A.y_B)^{h(m,r)} = r^r.g^s mod p, \tag{3.4}$$

where $r = r_A.r_B mod p$ and $s = s_A + s_B mod p - 1$. Thus the combined signature (r,s) is a valid contract signature of contract m.

3.3 Security discussion of the Existing Scheme

In this proposed scheme, the partial signatures(r_A, s_A) and (r_B, s_B) have no binding to any signer. Verifier cannot tell who is the signer because anyone, without knowing the private key x_A, can first randomly select s_A and solve for r_A to satisfy the equation $y_A^{h(m,r)} = r_A^r.g^{s_A} mod p$. This type of ambiguity, 1 out of ∞ ambiguity, can provide entirely ambiguous. On the other hand, the signature (r, s) binds to both signers since to generate a valid pair of (r, s) to

satisfy $(y_A.y_B)^{h(m,r)} = r^r.g^s modp$, needs to know the private key,$x = x_A + x_B$ of the public key.

3.4 Implementation result of Harn's scheme

The existing scheme is implemented in Java using BigInteger package. The signing and the verification phase has been shown.

```
Enter the approximate value of p you want.
45
Your prime p is 47.
Now, enter a number in between 2 and p-1 for generator g.
2
Person A: enter your private key xa now.
13
Person A: Calculates its public key ya=g^xa modp
Person B: enter your private number xb now.
17
Person B: Calculates its public key yb=g^xb modp
Person A: enter your secret number ka now.
19
Person A calculates ra=g^ka modp
Person A sends to person B ra=3.
Person B: enter your secret number kb now.
31
Person B calculates rb=g^kb modp
Person B sends to person A rb=21.
After receiving rb A computes r=ra.rb mod p
sa=xa.h(m,r)-ka.r mod p-1
A then sends sa to B
B verifies ya^h(m,r)=ra^r.g^sa mod p
LHS=21
RHS=21
It is equal so
B computes r=ra.rb mod p
sb=xa.h(m,r)-kb.r mod p-1
sb is sent to A
A verifies yb^h(m,r)=rb^r.g^sb mod p
LHS=28
RHS=28
It is equal so
Now both A and B can compute the contract signature (r,s) where s=sa+sb mod p-1 and r=ra.rb

C:\Users\sahaaakanksha7\Downloads\MY PROJECT\UGP>
```

Figure 3.1: Screen shot of the output.

Chapter 4

The Proposed Contract Signature Scheme

In this section, a contract signature scheme based on exchange protocol is proposed which allows two participants, namely, A and B, to interact with each other to sign a contact m over the Internet. The proposed scheme has two phases, Key generation and Exchange protocol. We assume that a contract m has been agreed between A and B before signing it. Moreover, it is assumed that the contract explicitly contains the following information: a predetermined but reasonable deadline, and the identities of A and B.

4.1 Key Generation

The operations of Key generation phase are described below.
In this phase the two parties (A and B) generate their private keys (x_a and x_b) and public keys (y_a and y_b), where, $y_a = g^{x_a} \mod p$ and $y_b = g^{x_b} \mod p$.

4.2 Exchange protocol

The operations of this phase are described below.
A computes its contract signature as:
$s_a = H(m, y_b^{s'} \mod p)$

Then sends s_a to B.

After receiving s_a, B checks its authenticity, by verifying the following condition.

$$s_a = H(m, (r_a^{x_b}.y_a^{rx_b}) \mod p) \tag{4.1}$$

If the above condition holds good, then B computes its contract signature as:

$s_b = H(m, y_b^{s''} \mod p)$

Then sends s_b to A.

After receiving s_b, A verifies if the following condition is true.

$$s_b = H(m, (r_b^{x_a}.y_b^{rx_a}) \mod p). \tag{4.2}$$

In this way both A and B compute their contract signature and execute the protocol successfully.

4.3 Implementation Results

Figure 4.1: Key Generation Phase

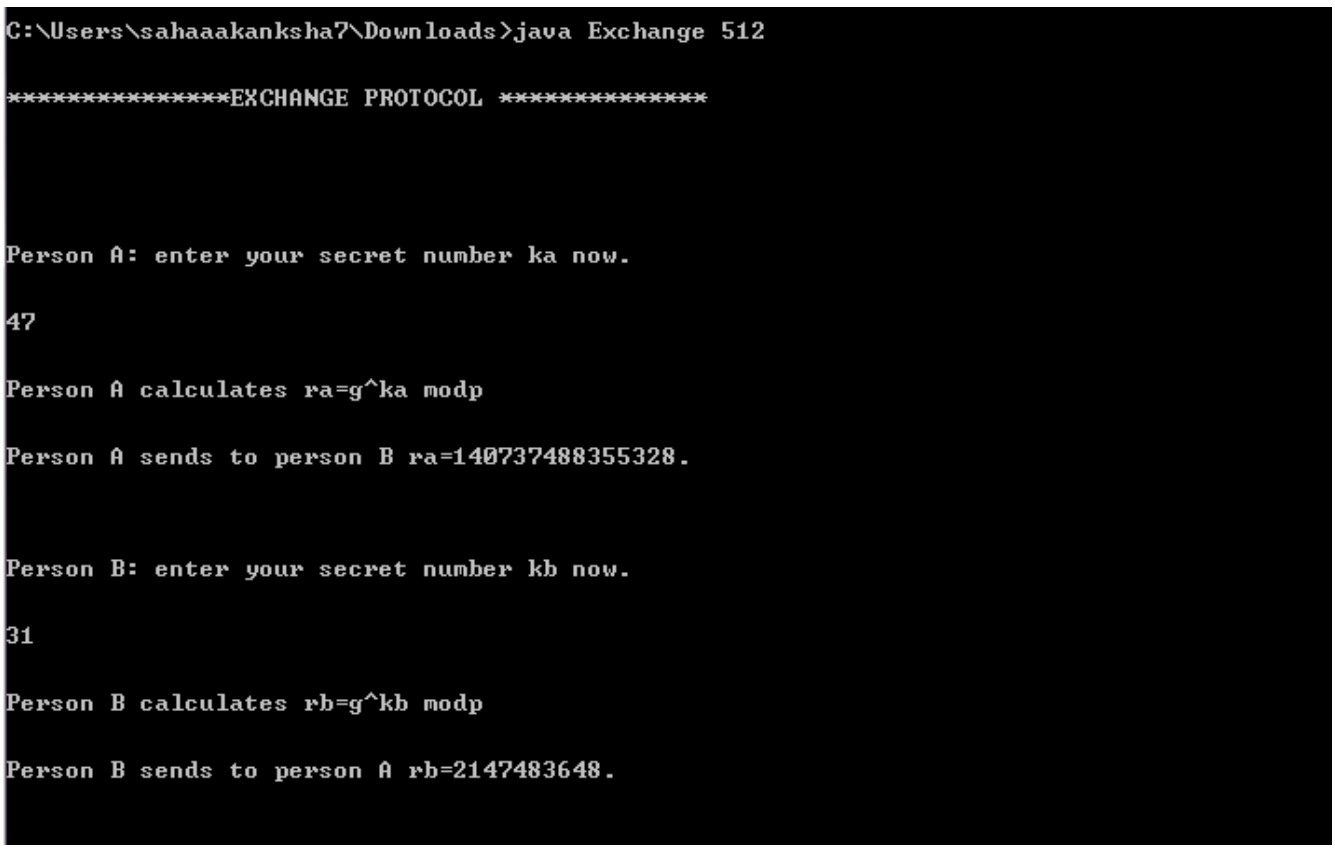

Figure 4.2: Exchange Protocol

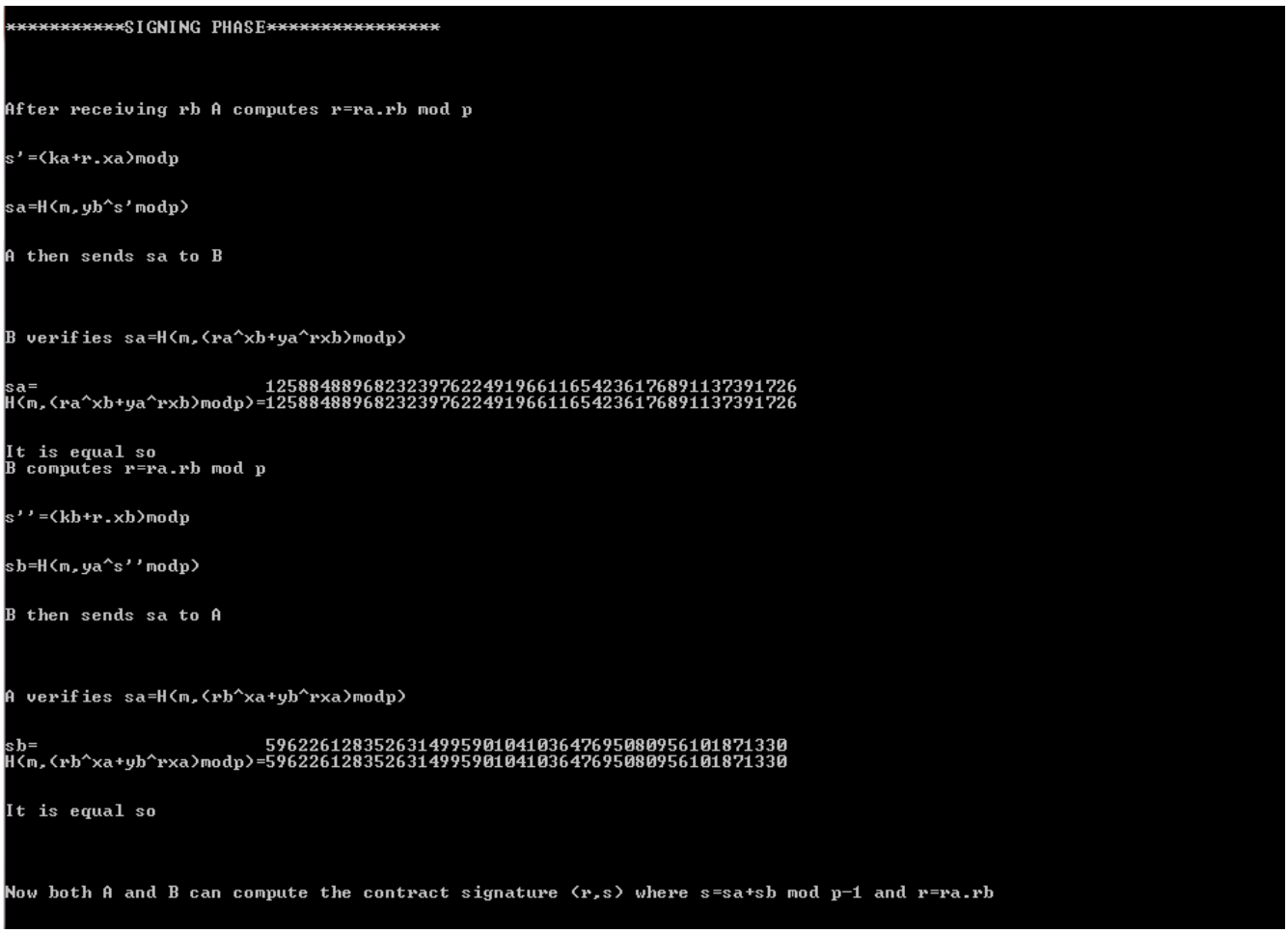

Figure 4.3: Signature Generation and Verification

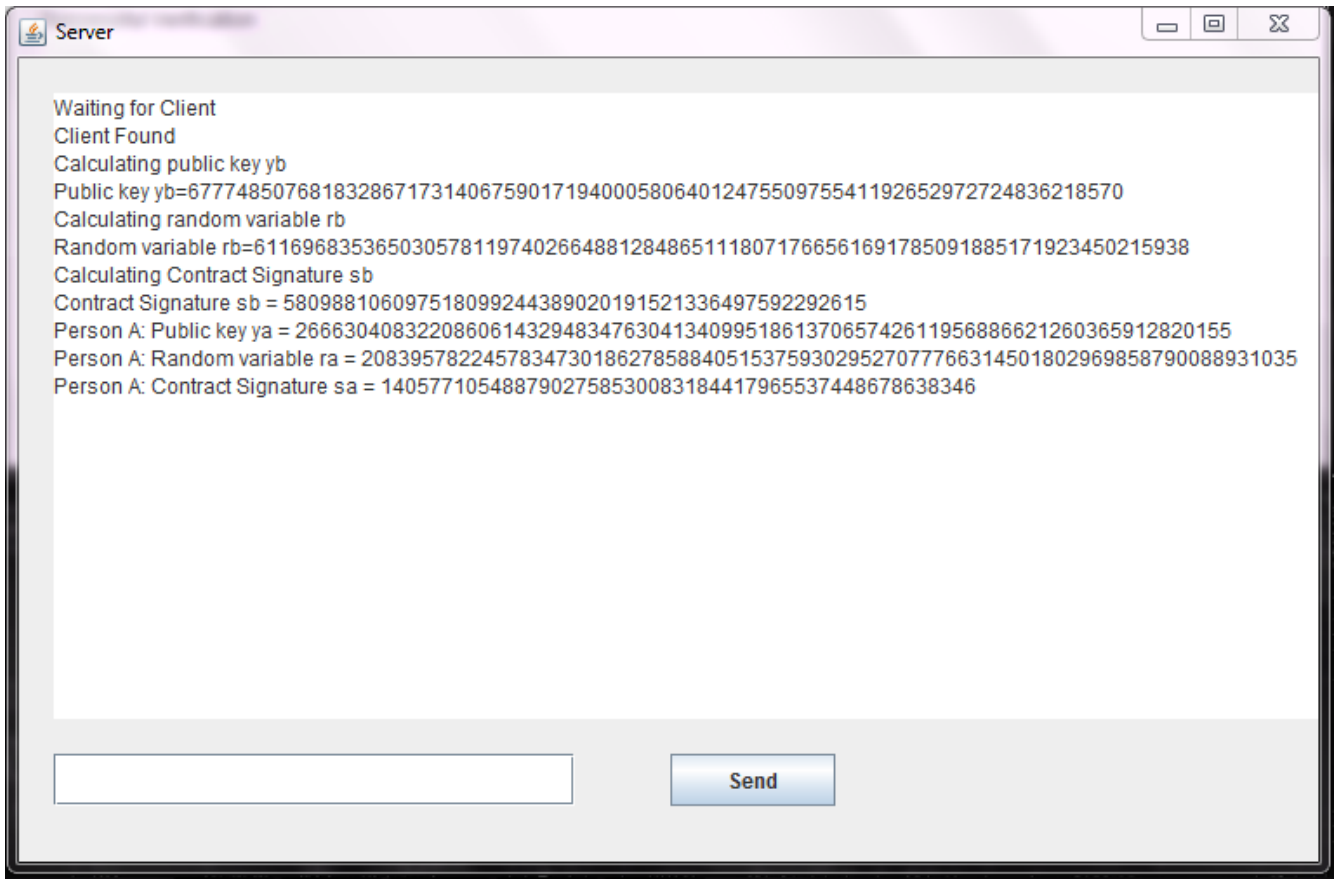

Figure 4.4: Contract Signature Protocol in Client Server Model: Server

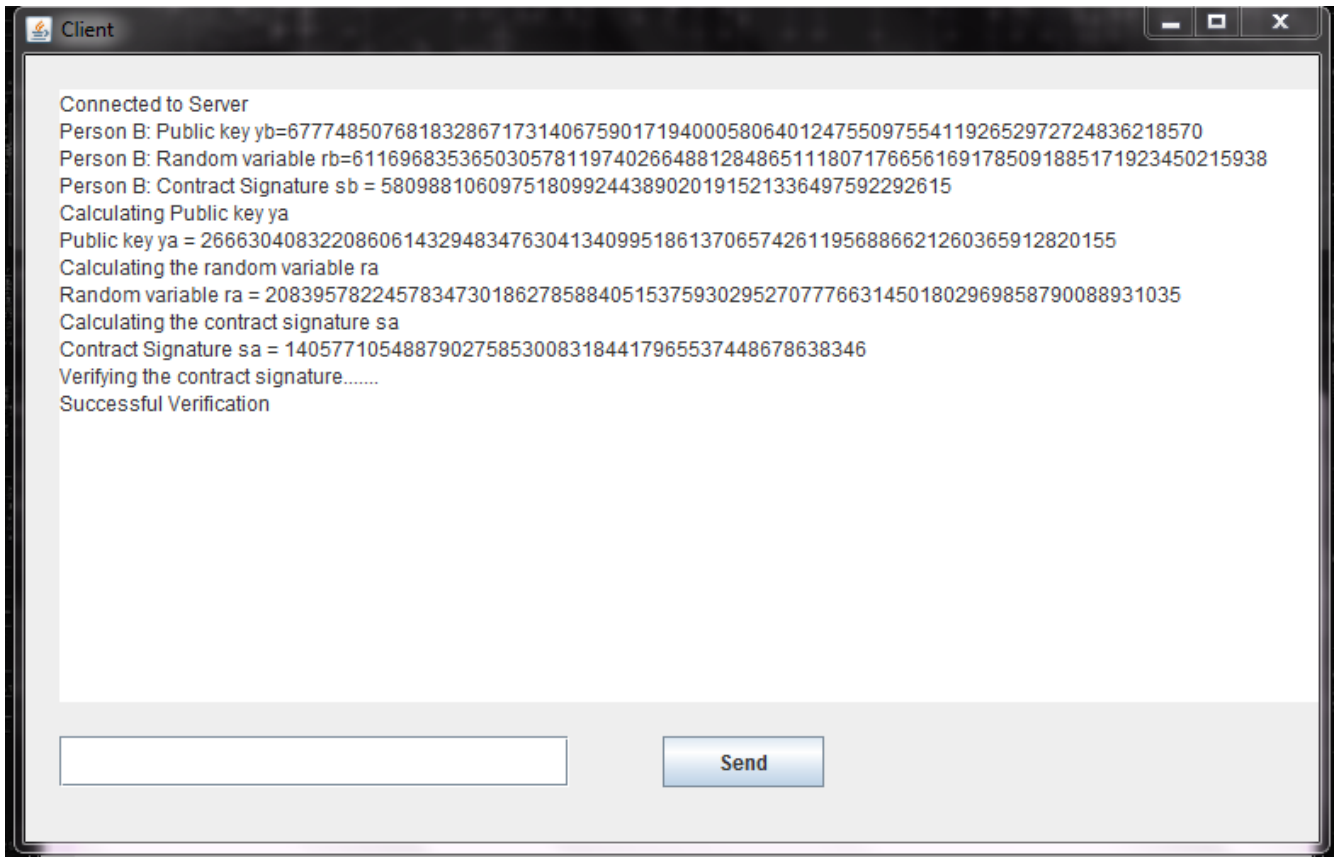

Figure 4.5: Contract Signature Protocol in Client Server Model: Client

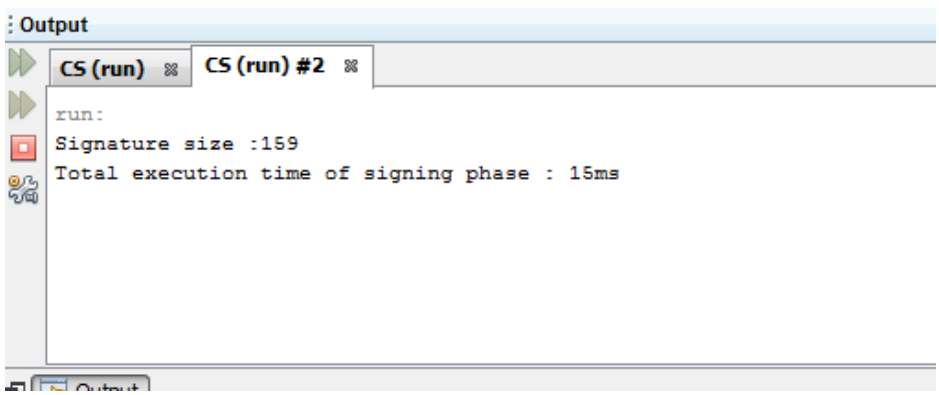

Figure 4.6: Length and Time for Signature Generation

4.4 Security Analysis of the Proposed Scheme

In this section, we first show the security assumptions of the proposed scheme. Then we discuss the correctness of the scheme. Subsequently, the performance of the proposed scheme is discussed.

Definition 1: Discrete logarithm problem (DLP). Let p be a large prime number and g be a generator of order $p-1$ in Z_p*. Given g, y and p, it is difficult to compute the exponent x from $y = g^x \mod p$ [2].

Correctness : The contract signature (r, s) of contract m is indeed correct.

The proofs are given below .

$$y_b^{s'} = g^{x_b(k_a + rx_a)} \tag{4.3}$$

$$= g^{x_a k_a} g^{x_b r x_a} \tag{4.4}$$

$$= r_a^{x_b} . y_a^{r x_b} \tag{4.5}$$

$$y_a^{s''} = g^{x_a(k_b + rx_b)} \tag{4.6}$$

$$= g^{x_b k_b} g^{x_b r x_a} \tag{4.7}$$

$$= r_b^{x_a} . y_b^{r x_a} \tag{4.8}$$

We compared the performance of the exchange protocol of the proposed scheme with Harn's scheme [1] and the result is tabulated in Table 4.1. The complexity of any signature scheme mostly depends on four operations, namely, exponentiation, multiplication, inverse operation and hash functions. In this evaluation, the time for performing modular addition and subtraction computations are ignored. The following notations are used to analyze the

performance of the schemes.

- T_E is the time for modular exponentiation
- T_M is the time for modular multiplication
- T_I is the time for modular inverse operation
- T_H is the time for performing hash functions

In our experiments we used $p = 256$ bits and the length of the signature generated is 159 bits.

Phase	**Harn's scheme [1]**	**Proposed scheme**
Exchange Protocol	$8T_E + 9T_M + 2T_H$	$8T_E + 7T_M + 2T_H$
Publicly verifiable	$3T_E + 2T_M + 1T_H$	$3T_E + 4T_M$

Table 4.1: Performance Comparison

Phase	**Harn's scheme [1]**	**Proposed scheme**
Signature and Verification	$26ms$	$15ms$

Table 4.2: Time Comparison (in millisecond)

It is observed from Table 4.1 and Table 4.2 that, the computational cost of the proposed scheme is considerably reduced as compared to Harn's scheme[1].

Chapter 5

Conclusion and Future Scope

In this thesis, we proposed a new digital contract signature scheme that allows two dishonest parties to exchange their signatures on a contract in an efficient and secure manner. The proposed protocol is abuse-free, i.e., if the signature protocol is not executed successfully, both the parties cannot show the validity of intermediate results generated by the other party to outsiders, during or after the procedure where those intermediate results are output. In other words, each party cannot convince an outsider to accept the partial commitments coming from the other party. This is an essential security property for contract signing, especially in the situations where partial commitments to a contract can be beneficial to a deceptive party or an outsider. Also, this scheme is registrant against some active attacks, such as forgery attack and adaptive chosen-message attack. Moreover, the proposed scheme satisfies some desirable security features, such as fairness, timely termination, compatibility, and high performance. The proposed scheme can also be used to implement an electronic transaction between two parties fairly. That is; it is guaranteed that the purchaser gets the digital goods from the store if and only if the store gets the money from the purchaser. In future research can be done on my scheme to lower its computation cost and communication overhead. Also, research can be done to incorporate contract signature protocol to highly security sensitive applications like e-bidding, e-voting, e-transactions etc.

Bibliography

[1] L. Harn and C. Lin,"Concurrent Signature in e-commerce,"*computers and electrical Engineering*, Vol.37, pp. 169-173, 2011. pp. 158-168, 2010.

[2] L. Harn, J. Ren and C. Lin.,"Design of DL based Certificateless Digital Signatures,"*Journal of Systems and Software*, Vol.82, pp. 789-793,2009.

[3] L. Chen, C. Kudla and KG. Paterson,"Concurrent signatures."*In: Proceedings of advances cryptology – EUROCRYPT '04*, Vol.3027, pp. 287-305, 2004.

[4] G.Wang,"An abuse free fair contract signing protocol based on RSA,"*IEEE Transaction on Information Forensics and Security*, Vol. 1, pp. 158-168, 2010.

[5] W. Susilo, Y. Mu and F. Zhang,"Perfect concurrent signature schemes," *In: Proceedings of the ICICS '04*, Vol.3269. pp. 14-26, 2004.

[6] K. Nguyen,"Asymmetric concurrent signatures,"*In: Proceedings of the ICICS'05*, Vol.3783. Berlin, Germany: Springer-Verlag;2005.

[7] L.Harn,"Group-oriented (t,n) threshold signature and multi signature," *IEE Proc-Comput Digital Tech*1994;141(5):307-13.

[8] S.Micali,"Simple and fast optimistic protocols for fair electronic exchange",*inProc. PODC'03*, pp. 12-19, ACM Press.

[9] T. ElGamal,"A public key crypto system and a signature scheme based on discrete logarithms,"*IEEE Trans.Inform. Theory*, Vol. IT-31, 1985

[10] Behrouz A. Forouzan. *Cryptography and network security.* Tata McGraw-Hill, 2007.

[11] William Stallings. *Cryptography and Network security: Principles and Practices.* Prentice Hall Inc., 1999.

[12] Goldwasser, Micali and Rivest, "A Digital signature scheme secure against adaptive chosen-message attacks." *SICOMP: SIAM Journal of Computing*,17, 1988.

[13] W. Diffie and M.E. Hellman, " New Directions in Cryptography." *IEEE Transactions on Information Theory* , 22(5):644-654, 1976.

[14] Chaum D., Fiat A., and Naor M. "Untraceable Electronic Cash" . *Springer-Verlag*, 319–327,1988.

[15] M. Abramowitz, I. A. Stegun, *Handbook of Mathematical Functions*, Applied Math. Series 55 (Tenth Printing ed.). National Bureau of Standards, p. 870, 1972

[16] Donald Knuth. *The Art of Computer Programming*, Volume 2: Seminumerical Algorithms, Third Edition. Addison-Wesley. Section 4.5.4: Factoring into Primes, pp. 379–417, 1997.

Dissemination

Sujata Mohanty, Aakanksha Saha, "A Novel Contract Signature based upon Key Exchange" in The Computer and Electrical Engineering, Elsevier.(Communicated on 5th March 2015)